WHAT IS CLAIMED IS:

1. A method for performing initialization operations in a system including a bus, bus interface, at least one bus device communicating on the bus, wherein the bus interface includes memory capable of being accessed over the bus by the at least one bus device, comprising:

detecting all bus devices capable of communicating on the bus;

configuring each detected bus device and bus interface with base addresses that enable transmission of Input/Output (I/O) requests over the bus to the memory in the bus interface and memory in any bus device including memory accessible over the bus;

testing the base addresses of the memory in each bus device including memory accessible over the bus by issuing I/O requests to the base addresses of the memory in each bus device; and

testing the memory in the bus interface by issuing I/O requests to the base addresses of the memory in the bus interface over the bus.

2. The method of claim 1, wherein the bus interface includes an initiator and target, wherein the memory in the bus interface is accessible through the target, and wherein testing the memory in the bus interface further comprises:

transmitting the I/O requests to the initiator, wherein the initiator transmits the I/O requests to the bus, and wherein the target accesses the I/O requests placed on the bus by the initiator and performs the requested I/O requests, whereby the I/O requests testing the memory on the bus interface further test circuitry connecting the target and the bus.

3. The method of claim 2, wherein the I/O requests to the memory of the bus interface comprise internal wrap signals between the initiator and target on the bus interface.

4. The method of claim 1, wherein the bus interface, bus, and bus devices implement the Peripheral Component Interconnect (PCI) architecture.

5. The method of claim 1, wherein the steps of detecting and configuring the bus devices, testing the base addresses of the memory, and testing the memory in the bus interface is performed by an initialization device embedded within the bus interface.

6. The method of claim 1, wherein the bus interface comprises a bridge between a primary bus and secondary bus, wherein detecting and configuring all the bus devices comprises detecting and configuring all the bus devices capable of communicating on the primary bus and the secondary bus.

7. The method of claim 6, wherein the steps of detecting and configuring the bus devices, testing the base addresses of the memory, and testing the memory in the bus interface is performed by an initialization device, wherein the initialization device communicates with the bridge over the primary bus.

8. The method of claim 7, wherein the bridge includes an initiator, a primary target to receive requests from the primary bus, and a secondary target to receive requests from the secondary bus, wherein the memory in the bridge is accessible through the secondary target, and wherein testing the memory in the bus interface further comprises:

transmitting the I/O requests to the primary target over the primary bus, wherein the primary target transmits the I/O requests to the initiator, and wherein the initiator sends the I/O requests to the secondary bus, wherein the secondary target accesses the I/O requests placed on the secondary bus by the initiator and performs the requested I/O requests to the base addresses of the memory in the bridge, whereby the I/O requests testing the memory in the bridge tests circuitry connecting the secondary target and the secondary bus.

9. The method of claim 8, wherein the I/O requests to the bridge comprise internal wrap signals between the initiator and the secondary target in the bridge.

10. The method of claim 1, wherein testing the memory in the bus interface and each bus device further comprises:

writing data to the base addresses of the memory;

reading the base addresses to which data was written; and

for each base address to which data was written, comparing the data read with the data written to determine whether the data read and written to each base address is the same, wherein there is an error if the data read and written data does not match.

11. The method of claim 1, wherein a Direct Memory Access (DMA) engine performs the step of testing the memory in the bus interface and wherein an initialization device performs the steps of detecting and configuring the bus devices.

12. A method for performing a verification of a bus interface including an embedded device and memory, wherein the bus interface enables communication with a bus, wherein the bus interface memory is capable of being accessed by one bus device communicating over the bus, and wherein the embedded device uses the bus interface to communicate on the bus, comprising:

causing the bus device to test the memory in the bus interface by issuing Input/Output (I/O) requests to the memory in the bus interface over the bus; and

causing the embedded device to test the memory in the bus interface by issuing Input/Output (I/O) requests to the memory in the bus interface over the bus, whereby the tests performed by the bus device and embedded device test whether the bus interface is capable of handling requests from multiple bus devices over the bus.

13. The method of claim 12, wherein the bus interface includes an initiator and target, wherein the memory in the bus interface is accessible through the target, and wherein testing the memory in the bus interface with the embedded device further comprises:

transmitting the I/O requests to the initiator, wherein the initiator transmits the I/O requests to the bus, and wherein the target accesses the I/O requests placed on the bus by the initiator and performs the requested I/O requests, whereby the I/O requests

from the initiator testing the memory on the bus interface further test circuitry connecting the target and bus.

14. The method of claim 13, wherein the I/O requests to the memory of the bus interface from the embedded device comprises internal wrap signals between the initiator and target on the bus interface.

15. The method of claim 12, wherein the bus interface, bus, and bus devices implement the Peripheral Component Interconnect (PCI) architecture.

16. The method of claim 12, wherein the embedded device further performs initialization operations to assign base addresses to bus devices and the bus interface that communicate over the bus.

17. A system for performing initialization operations, comprising:

(a) a bus;

(b) a bus interface;

(c) at least one bus device communicating on the bus;

(d) memory within the bus interface capable of being accessed over the bus by the at least one bus device;

(e) an initialization device that uses the bus interface to communicate on the bus;

(f) logic implemented in the initialization device to initialize communication on the bus by performing:

(i) detecting all bus devices capable of communicating on the bus;

(ii) configuring each detected bus device and bus interface with base addresses that enable transmission of Input/Output (I/O) requests over the bus to the memory in the bus interface and memory in any bus device including memory accessible over the bus;

(iii) testing the base addresses of the memory in each bus device including memory accessible over the bus by issuing I/O requests to the base addresses of the memory in each bus device; and

(iv) testing the memory in the bus interface by issuing I/O requests to the base addresses of the memory in the bus interface over the bus.

18. The system of claim 17, further comprising:

an initiator and target in the bus interface, wherein the memory in the bus interface is accessible through the target, and wherein the logic implemented in the initialization device further performs:

transmitting the I/O requests to the initiator, wherein the initiator transmits the I/O requests to the bus, and wherein the target accesses the I/O requests placed on the bus by the initiator and performs the requested I/O requests, whereby the I/O requests testing the memory on the bus interface further test circuitry connecting the target and the bus.

19. The system of claim 18, wherein the I/O requests to the memory of the bus interface comprise internal wrap signals between the initiator and target on the bus interface.

20. The system of claim 17, wherein the bus interface, bus, bus devices, and initialization device implement the Peripheral Component Interconnect (PCI) architecture.

21. The system of claim 17, wherein the initialization device is embedded within the bus interface.

22. The system of claim 17, further comprising:

a primary bus;

a secondary bus, wherein the bus interface comprises a bridge between the primary bus and secondary bus, wherein the logic implemented in the initialization

device for detecting and configuring all the bus devices detects and configures all the bus devices capable of communicating on the primary bus and the secondary bus.

23. The system of claim 22, wherein the initialization device communicates with the bridge over the primary bus.

24. The system of claim 23, further comprising:

a primary target in the bridge to receive requests from the primary bus;

a secondary target in the bridge to receive requests from the secondary bus, wherein the memory in the bridge is accessible through the secondary target;

an initiator in the bridge, and

wherein the logic implemented in the initialization device for testing the memory in the bus interface further transmits the I/O requests to the primary target over the primary bus, wherein the primary target transmits the I/O requests to the initiator, and wherein the initiator sends the I/O requests to the bus, wherein the secondary target accesses the I/O requests placed on the secondary bus by the initiator and performs the requested I/O requests to the base addresses of the memory in the bridge, whereby the I/O requests testing the memory in the bridge tests circuitry connecting the secondary target and the secondary bus.

25. The system of claim 24, wherein the I/O requests to the bridge comprise internal wrap signals between the initiator and the secondary target in the bridge.

26. The system of claim 17, wherein the logic implemented in the initialization device for testing the memory in the bus interface and each bus device further performs:

writing data to the base addresses of the memory;

reading the base addresses to which data was written; and

for each base address to which data was written, comparing the data read with the data written to determine whether the data read and written to each base address is the same, wherein there is an error if the data read and written data does not match.

27. The system of claim 17, further comprising:

a Direct Memory Access (DMA) engine, wherein the initialization device transmits the DMA engine the writes to test the memory in the bus interface.

28. A system for performing a verification of a bus interface, comprising:

a bus interface;

an embedded device within the bus interface;

a memory within the bus interface;

a bus, wherein the bus interface enables communication with the bus;

at least one bus device capable of accessing the bus interface memory over the bus, wherein the embedded device uses the bus interface to communicate on the bus;

logic implemented in one bus device to test the memory in the bus interface by issuing Input/Output (I/O) requests to the memory in the bus interface over the bus; and

logic implemented in the embedded device to test the memory in the bus interface by issuing Input/Output (I/O) requests to the memory in the bus interface over the bus, whereby the tests performed by the bus device and embedded device test whether the bus interface is capable of handling requests from multiple bus devices over the bus.

29. The system of claim 28, further comprising:

an initiator in the bus interface;

a target in the bus interface, wherein the memory in the bus interface is accessible through the target, and

wherein the logic implemented in the embedded device for testing the memory in the bus interface further transmits the I/O requests to the initiator, wherein the initiator transmits the I/O requests to the bus, and wherein the target accesses the

I/O requests placed on the bus by the initiator and performs the requested I/O requests, whereby the I/O requests from the initiator testing the memory on the bus interface further test circuitry connecting the target and bus.

30. The system of claim 28, wherein the I/O requests to the memory of the bus interface from the embedded device comprises internal wrap signals between the initiator and target on the bus interface.

31. The system of claim 28, wherein the bus interface, bus, and bus devices implement the Peripheral Component Interconnect (PCI) architecture.

32. The system of claim 28, wherein the embedded device further includes logic to perform initialization operations to assign base addresses to bus devices and the bus interface that communicate over the bus.

33. An article of manufacture including code for performing initialization operations in a system including a bus, bus interface, at least one bus device communicating on the bus, wherein the bus interface includes memory capable of being accessed over the bus by the at least one bus device, wherein the code causes operations comprising:

detecting all bus devices capable of communicating on the bus;

configuring each detected bus device and bus interface with base addresses that enable transmission of Input/Output (I/O) requests over the bus to the memory in the bus interface and memory in any bus device including memory accessible over the bus;

testing the base addresses of the memory in each bus device including memory accessible over the bus by issuing I/O requests to the base addresses of the memory in each bus device; and

testing the memory in the bus interface by issuing I/O requests to the base addresses of the memory in the bus interface over the bus.

34. The article of manufacture of claim 33, wherein the bus interface includes an initiator and target, wherein the memory in the bus interface is accessible through the target, and wherein testing the memory in the bus interface further comprises:

transmitting the I/O requests to the initiator, wherein the initiator transmits the I/O requests to the bus, and wherein the target accesses the I/O requests placed on the bus by the initiator and performs the requested I/O requests, whereby the I/O requests testing the memory on the bus interface further test circuitry connecting the target and the bus.

35. The article of manufacture of claim 34, wherein the I/O requests to the memory of the bus interface comprise internal wrap signals between the initiator and target on the bus interface.

36. The article of manufacture of claim 33, wherein the bus interface comprises a bridge between a primary bus and secondary bus, wherein detecting and configuring all the bus devices comprises detecting and configuring all the bus devices capable of communicating on the primary bus and the secondary bus.

37. The article of manufacture of claim 36, wherein the bridge includes an initiator, a primary target to receive requests from the primary bus, and a secondary target to receive requests from the secondary bus, wherein the memory in the bridge is accessible through the secondary target, and wherein testing the memory in the bus interface further comprises:

transmitting the I/O requests to the primary target over the primary bus, wherein the primary target transmits the I/O requests to the initiator, and wherein the initiator sends the I/O requests to the bus, wherein the secondary target accesses the I/O requests placed on the secondary bus by the initiator and performs the requested I/O requests to the base addresses of the memory in the bridge, whereby the I/O requests testing the memory in the bridge tests circuitry connecting the secondary target and the secondary bus.

38. The article of manufacture of claim 37, wherein the I/O requests to the bridge comprise internal wrap signals between the initiator and the secondary target in the bridge.

39. The article of manufacture of claim 33, wherein testing the memory in the bus interface and each bus device further comprises:

writing data to the base addresses of the memory;

reading the base addresses to which data was written; and

for each base address to which data was written, comparing the data read with the data written to determine whether the data read and written to each base address is the same, wherein there is an error if the data read and written data does not match.

40. An article of manufacture of code for performing a verification of a bus interface including an embedded device and memory, wherein the bus interface enables communication with a bus, wherein the bus interface memory is capable of being accessed by one bus device communicating over the bus, and wherein the embedded device uses the bus interface to communicate on the bus, wherein the code causes operations comprising:

causing the bus device to test the memory in the bus interface by issuing Input/Output (I/O) requests to the memory in the bus interface over the bus; and

causing the embedded device to test the memory in the bus interface by issuing Input/Output (I/O) requests to the memory in the bus interface over the bus, whereby the tests performed by the bus device and embedded device test whether the bus interface is capable of handling requests from multiple bus devices over the bus.

41. The article of manufacture of claim 40, wherein the bus interface includes an initiator and target, wherein the memory in the bus interface is accessible through the target, and wherein testing the memory in the bus interface with the embedded device further comprises:

transmitting the I/O requests to the initiator, wherein the initiator transmits the I/O requests to the bus, and wherein the target accesses the I/O requests placed on the

bus by the initiator and performs the requested I/O requests, whereby the I/O requests from the initiator testing the memory on the bus interface further test circuitry connecting the target and bus.

42. The article of manufacture of claim 41, wherein the I/O requests to the memory of the bus interface from the embedded device comprises internal wrap signals between the initiator and target on the bus interface.

43. A system for performing initialization operations in a system including a bus, bus interface, at least one bus device communicating on the bus, wherein the bus interface includes memory capable of being accessed over the bus by the at least one bus device, comprising:

means for detecting all bus devices capable of communicating on the bus;

means for configuring each detected bus device and bus interface with base addresses that enable transmission of Input/Output (I/O) requests over the bus to the memory in the bus interface and memory in any bus device including memory accessible over the bus;

means for testing the base addresses of the memory in each bus device including memory accessible over the bus by issuing I/O requests to the base addresses of the memory in each bus device; and

means for testing the memory in the bus interface by issuing I/O requests to the base addresses of the memory in the bus interface over the bus.

44. The system of claim 43, wherein the bus interface comprises a bridge between a primary bus and secondary bus, wherein the means for detecting and configuring all the bus devices detects and configures all the bus devices capable of communicating on the primary bus and the secondary bus.

45. The system of claim 44, wherein the bridge includes a primary target to receive requests from the primary bus, a secondary target to receive requests from the secondary bus, wherein the memory in the bridge is accessible through the

secondary target, and wherein the means for testing the memory in the bus interface further performs:

transmitting the I/O requests to the primary target over the primary bus, wherein the primary target transmits the I/O requests to the initiator, and wherein the initiator sends the I/O requests to the bus, wherein the secondary target accesses the I/O requests placed on the secondary bus by the initiator and performs the requested I/O requests to the base addresses of the memory in the bridge, whereby the I/O requests testing the memory in the bridge tests circuitry connecting the secondary target and the secondary bus.

46. The system of claim 45, wherein the I/O requests to the bridge comprise internal wrap signals between the initiator and the secondary target in the bridge.

47. The system of claim 43, wherein the means for testing the memory in the bus interface and each bus device further performs:

writing data to the base addresses of the memory;

reading the base addresses to which data was written; and

for each base address to which data was written, comparing the data read with the data written to determine whether the data read and written to each base address is the same, wherein there is an error if the data read and written data does not match.

48. A system for performing a verification of a bus interface, comprising:

a bus interface including an embedded device and memory;

a bus, wherein the bus interface enables communication with the bus;

a bus device communicating over the bus, wherein the bus interface memory is capable of being accessed by one bus device communicating over the bus, and wherein the embedded device uses the bus interface to communicate on the bus;

means for causing the bus device to test the memory in the bus interface by issuing Input/Output (I/O) requests to the memory in the bus interface over the bus; and

means for causing the embedded device to test the memory in the bus interface by issuing Input/Output (I/O) requests to the memory in the bus interface over the bus, whereby the tests performed by the bus device and embedded device test whether the bus interface is capable of handling requests from multiple bus devices over the bus.

49. The system of claim 48, wherein the bus interface includes an initiator and target, wherein the memory in the bus interface is accessible through the target, and wherein the means for testing the memory in the bus interface with the embedded device further performs:

transmitting the I/O requests to the initiator, wherein the initiator transmits the I/O requests to the bus, and wherein the target accesses the I/O requests placed on the bus by the initiator and performs the requested I/O requests, whereby the I/O requests from the initiator testing the memory on the bus interface further test circuitry connecting the target and bus.